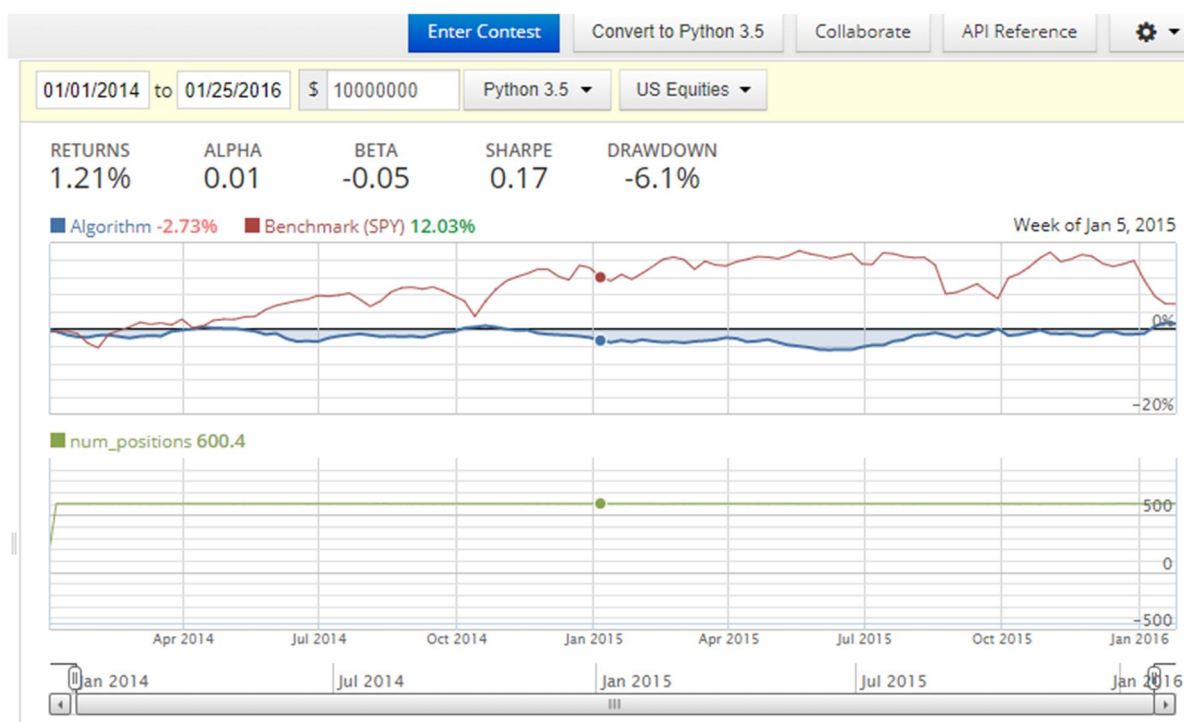


Algoritmo de Trading

Tomando el algoritmo Cross-sectional Equity Template, de

<https://www.quantopian.com/algorithms/5e40bbf686d3f6003e89c67c> , se obtiene

un resultado similar a este:



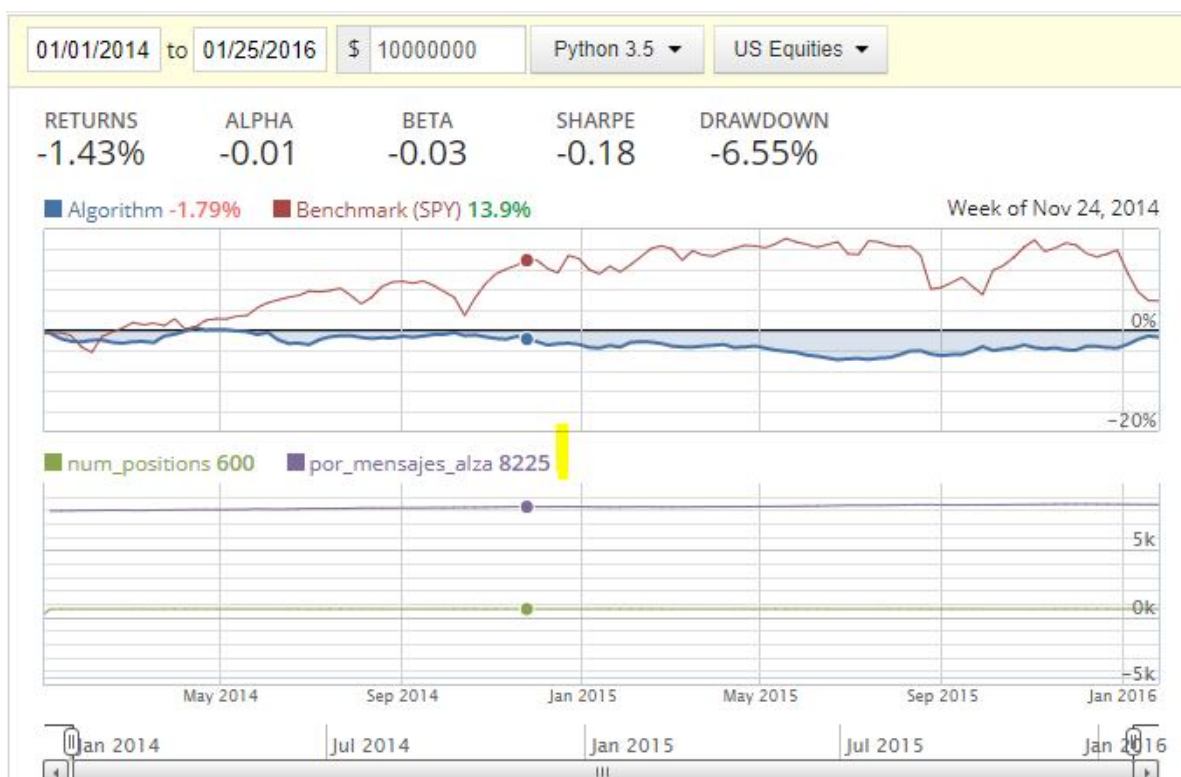
Este algoritmo tiene originalmente 3 variables, se le incluyen 3 variables más:

```
#Obtenemos el ultimo precio de cierre del dia, de todas las acciones
yesterday_close = EquityPricing.close.latest
#Obtenemos el ultimo volumen de operaciones diarias, para todas las acciones
yesterday_volume = EquityPricing.volume.latest
#Obtiene el conjunto de datos para los mensajes de twitter, incluyendo re-tweets
tweets_bear = twitter_withretweets.bear_scored_messages.latest

|
# Here we combine our winsorized factors, z-scoring them to equalize their influence
combined_factor = (
    value_winsorized.zscore() +
    quality_winsorized.zscore() +
    sentiment_score_winsorized.zscore()+
    yesterday_close.zscore()+
    yesterday_volume.zscore()+
    tweets_bear.zscore()
)
```

- **yesterday_close:** Obtenemos el ultimo precio de cierre del día, de todas las acciones usando EquityPricing.close.latest.
- **yesterday_volume:** Obtenemos el ultimo volumen de operaciones diarias, para todas las acciones. Usamos EquityPricing.volume.latest
- **tweets_bear:** Obtiene el conjunto de datos para los mensajes de twitter, incluyendo re-tweets. Usamos los re-tweets, ya que indica que llego a más gente el mensaje. twitter_withretweets.bear_scored_messages.latest

Se puede notar que el algoritmo varia, respecto al original. También, aparece otra grafica, con el label “por_mensajes_alza”



La relación se considera muy alta. Se recomienda revisar muy bien para futuros desarrollos.

Esta corresponde al relación de mensajes relacionados sentimientos de alza, sobre el total de mensajes de twitter.

```
def make_pipeline2():
    """
    Esta funcion retorna un Pipeline con informacion tomada de redes sociales
    la cual ha sido procesada en lenguaje natural para determinar poder
    cuantificar los sentimientos. En este caso, se usa informacion de twitter
    """
    pipe = Pipeline( columns={
        'tweets': twitter_withretweets.bull_scored_messages.latest /
            twitter_withretweets.total_scanned_messages.latest

    })
    return pipe

def record_vars2(context, data):
    """
    se obtiene el la relacion entre mensajes relacionados con el alza,
    sobre el total de mensajes
    """
    context.datos_test = algo.pipeline_output('cantidadTweets')
    algo.record(por_mensajes_alza=len(context.datos_test))
```

Referencias:

<https://www.quantopian.com/docs/data-reference/psychsignal>

https://www.quantopian.com/docs/data-reference/daily_pricing